

Head and Neck Tumor and Lymph Node Segmentation and Outcome Prediction from 18F-FDG PET/CT Images: Simplicity is All You Need

Louis Rebaud^{1 2}, Thibault Escobar^{1 3},
Fahad Khalid¹, Kibrom Girum¹
and Irène Buvat¹

¹ Laboratory of Translational Imaging in Oncology (LITO), U1288,
Institut Curie, Inserm, Université Paris-Saclay, Orsay, France

² Siemens Healthcare SAS, Saint Denis, France

³ DOSIsoft SA, Cachan, France

Disclosure

- **Louis Rebaud** is an employee of **Siemens Healthineers**
- **Thibault Escobar** is an employee of **DOSIsoft**



Team



Louis Rebaud
2nd year PhD student



Thibault Escobar
3rd year PhD student



Irène Buvat
Head of the LITO lab



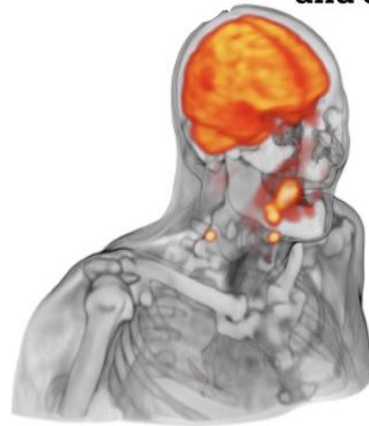
Fahad Khalid
3rd year PhD student



Kibrom Girum
Postdoctoral researcher

Summary

- Data
- Proposed automatic pipeline
- Segmentation (Task 1)
- Outcome prediction (Task 2)
- Conclusion



HECKTOR 2022

HEAd and neCK TumOR segmentation
and outcome prediction in PET/CT images
Third edition

MICCAI2022
Singapore

THE UNIVERSITY OF TEXAS
MD Anderson
Cancer Center

CHUV HUG

Hôpitaux
Universitaires
Genève

USZ Universitäts
Spital Zürich
BRIGHAM AND
WOMEN'S HOSPITAL

CHU
de Poitiers

CENTRE du Sud-Centre de Cancer
HENRI
BECQUEREL
unijancer NORMANDIE - ROUEN

Centre
Eugène Marquis
RENNES

Hes·SO VALAIS
WALLIS

Σ π ≈ &

Inserm
La science pour la santé
From science to health
UBO
Université de Bretagne Occidentale

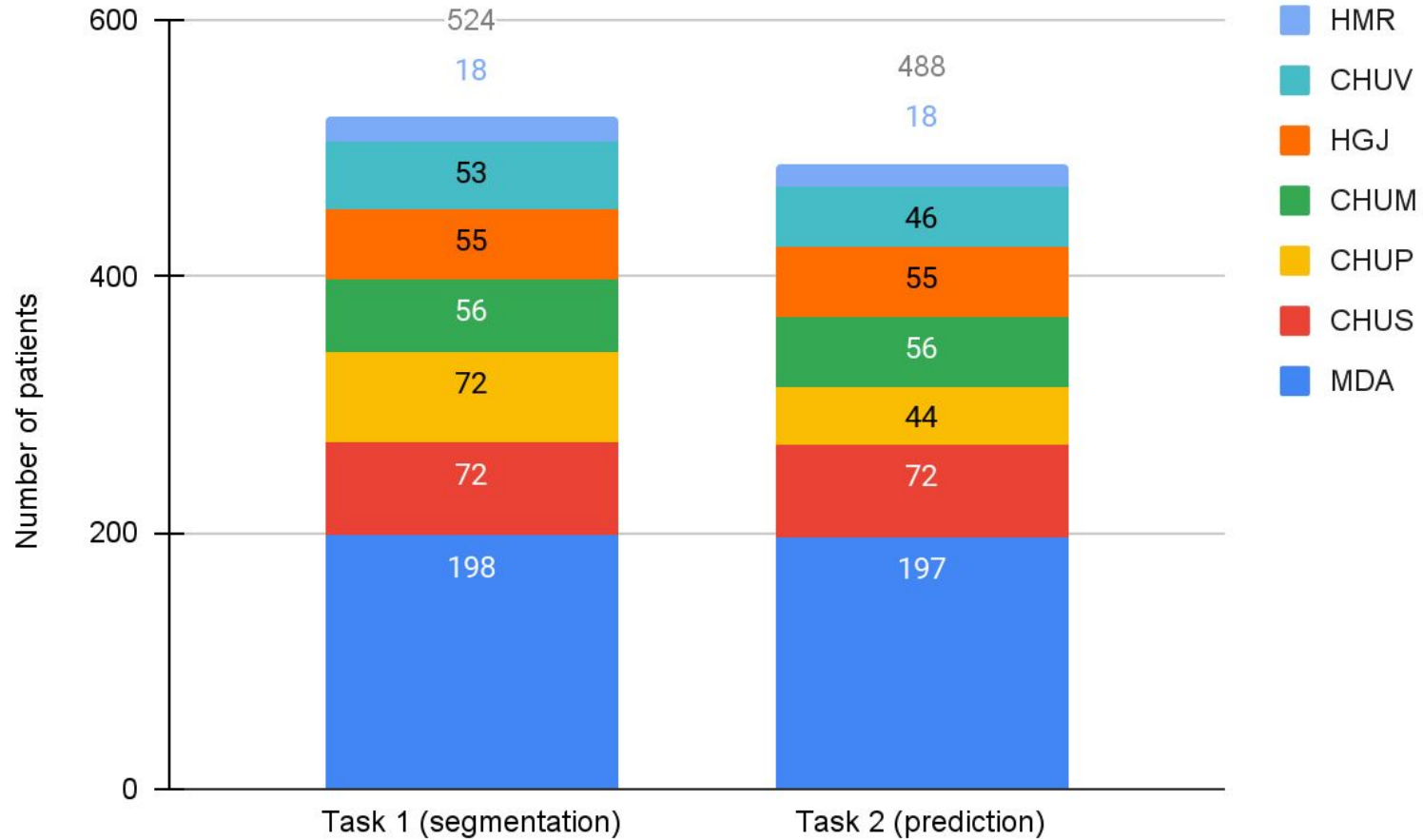
LATIM

UNIVERSITÉ
DE ROUEN
NORMANDIE

litis

UNIVERSITÉ DE
SHERBROOKE

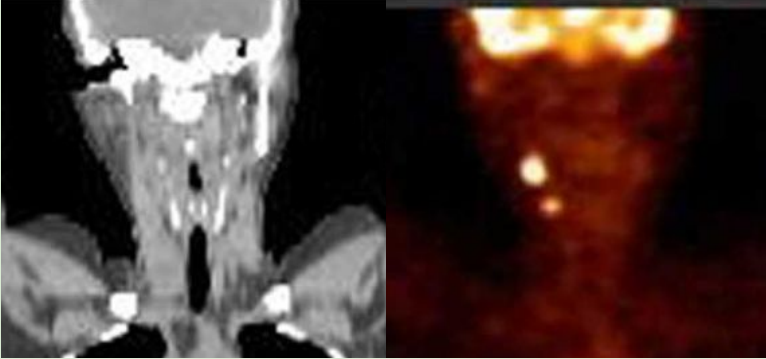
Data - number of patients



Data - modalities

FDG PET/CT images

524/524
patients



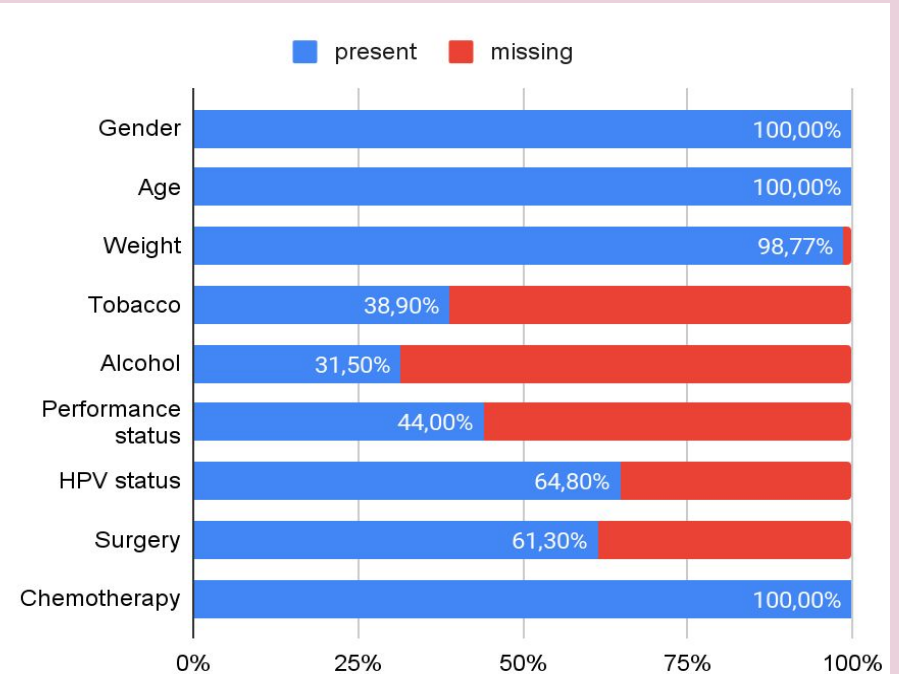
Outcome data

488/524
patients

- Recurrence free survival (time)
- Relapse (binary)

Clinical data

488/524
patients



Proposed automated pipeline

18F-FDG PET/CT images

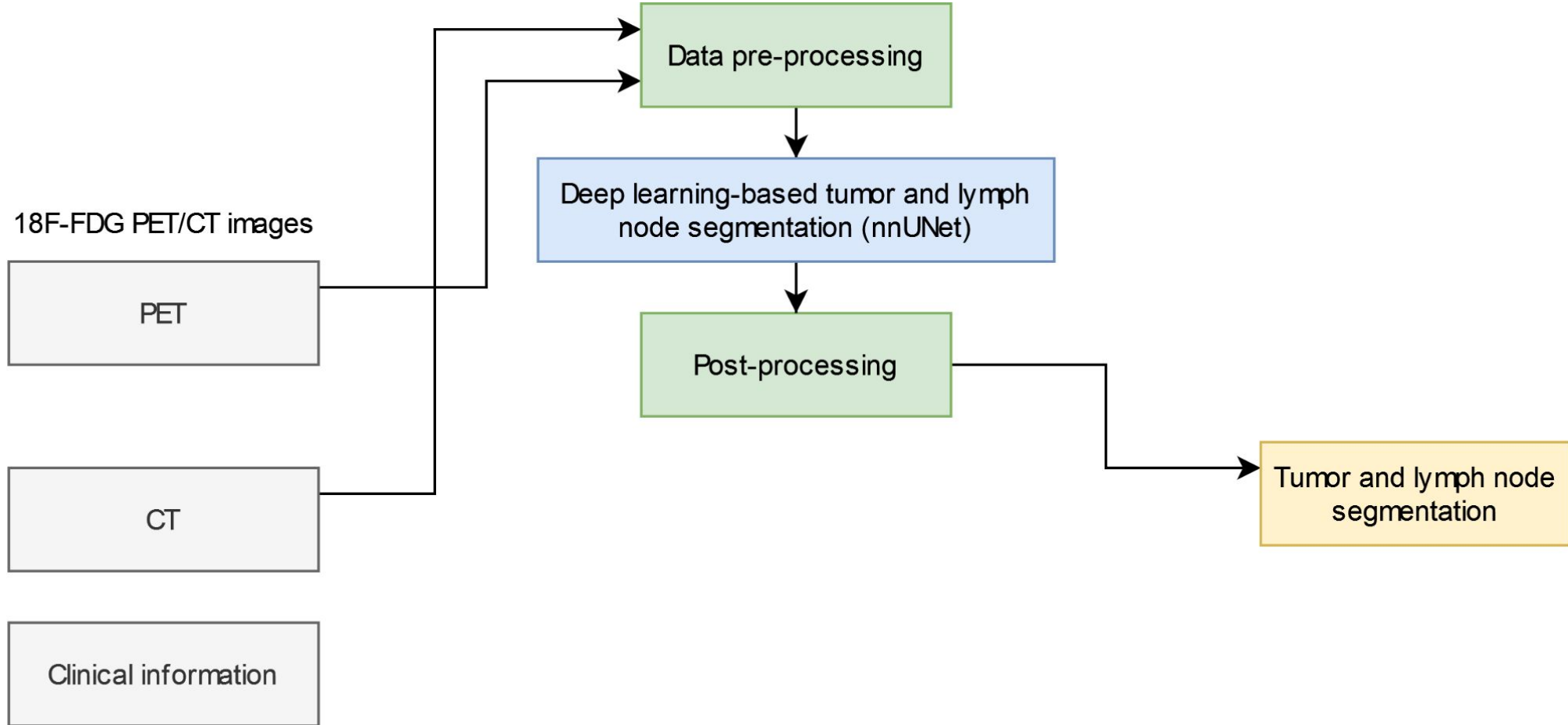


PET

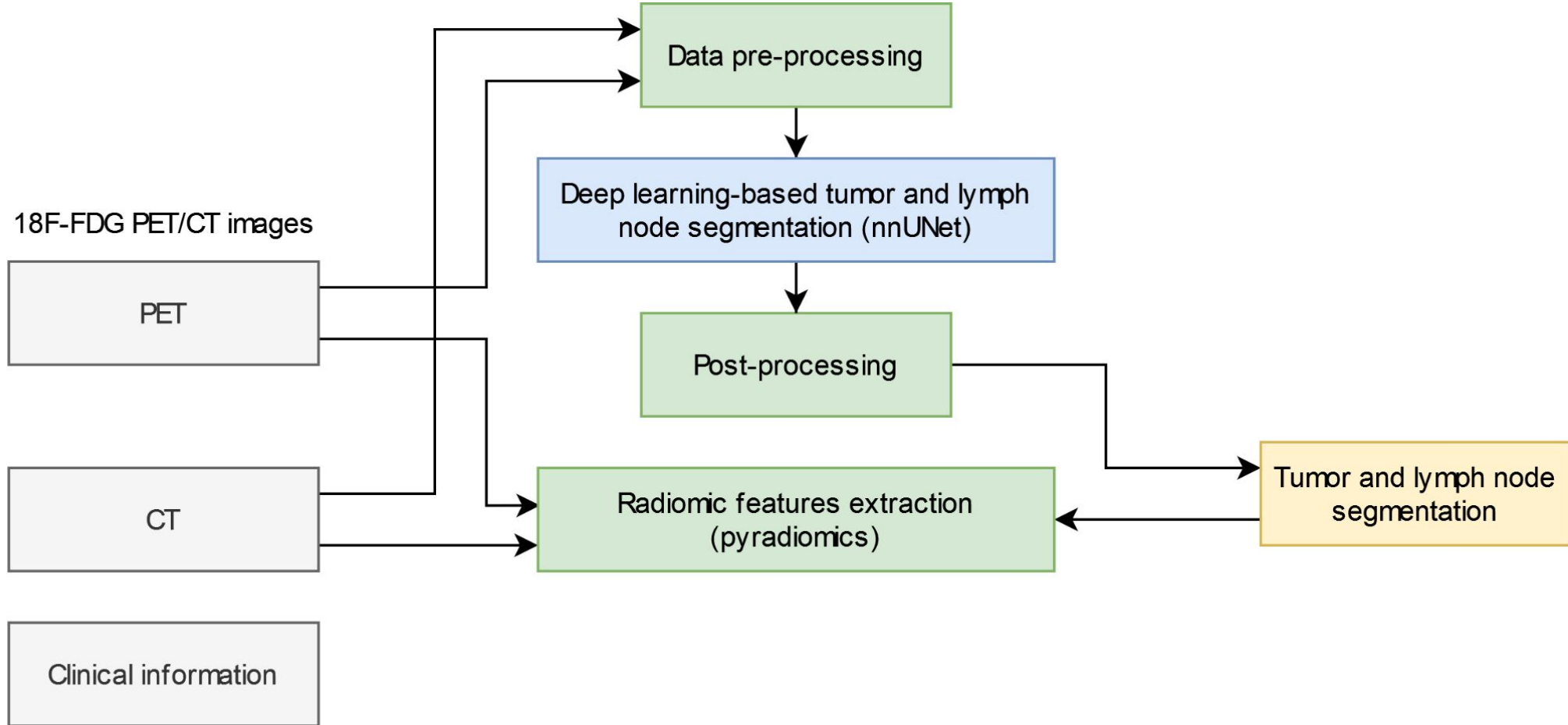
CT

Clinical information

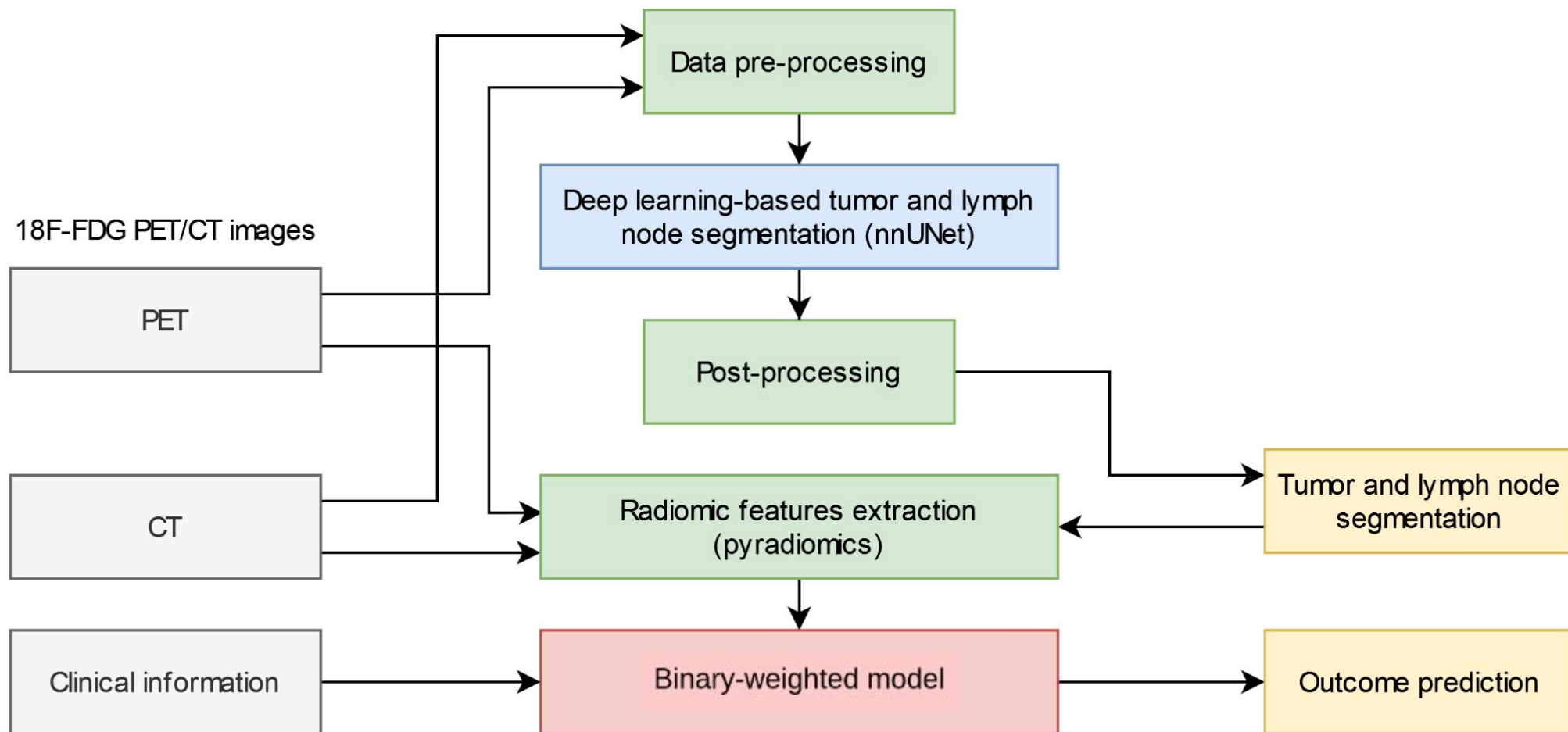
Proposed automated pipeline



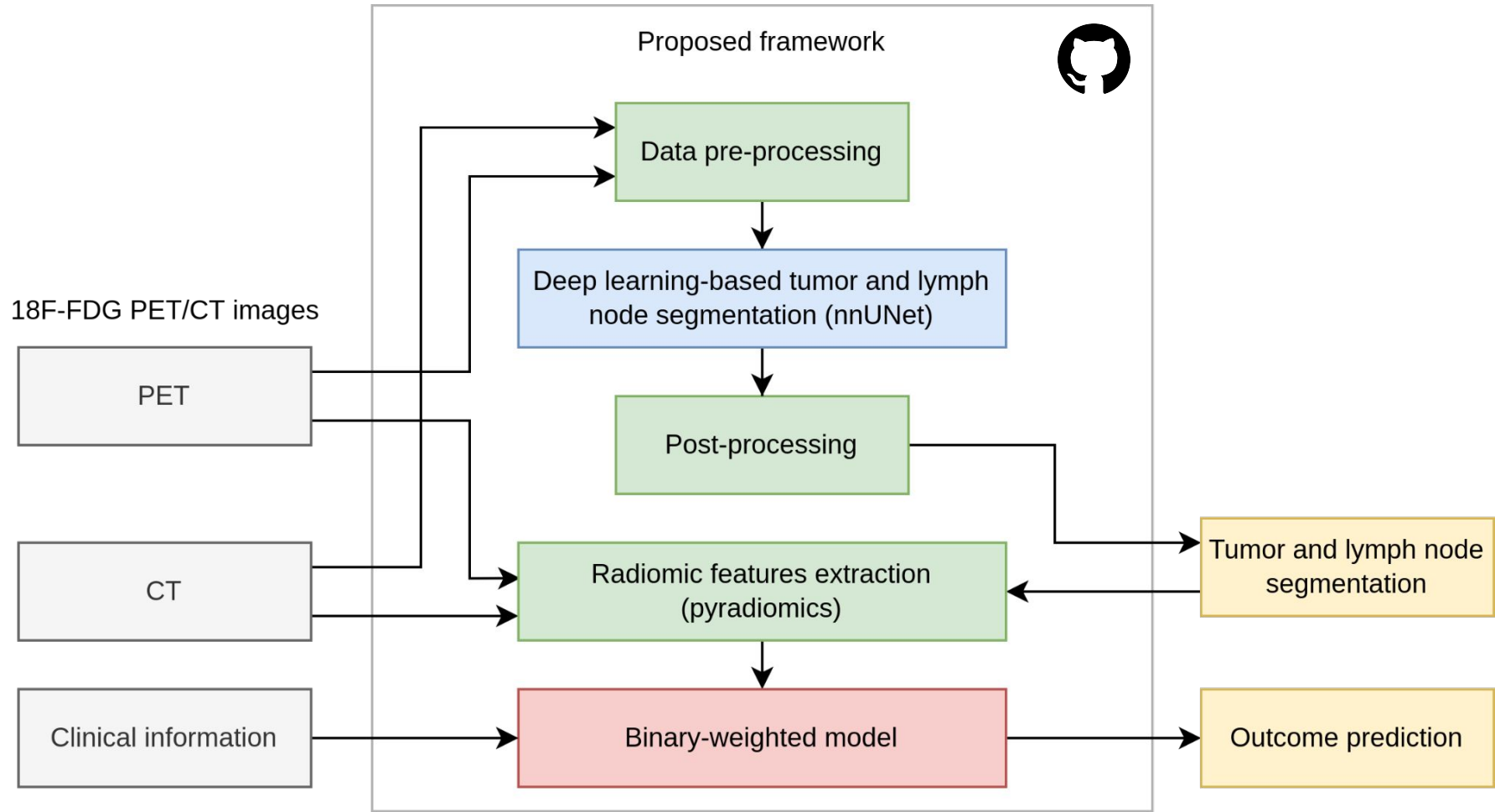
Proposed automated pipeline



Proposed automated pipeline

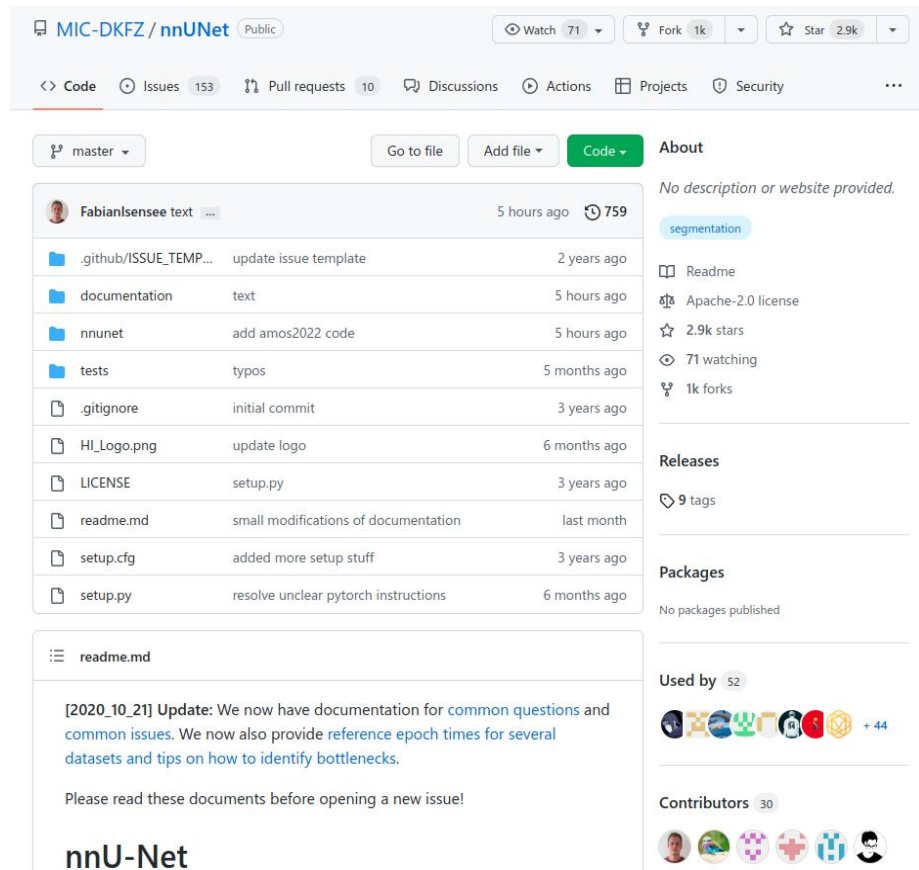


Proposed automated pipeline



Segmentation (Task 1) - Method

- **Out-of-the-box nnUNet** in “3D full resolution” mode
- Preprocessing: **2mm × 2mm × 2mm resampling** of PET and CT images (spline) and ground truth masks (nearest neighbor)
- Other **preprocessing** and **training parameters by default**
- Postprocessing: **median filter** with a $3 \times 3 \times 3$ voxel kernel size applied to the output resampled to the initial voxel size, to **smooth out the staircase effect**



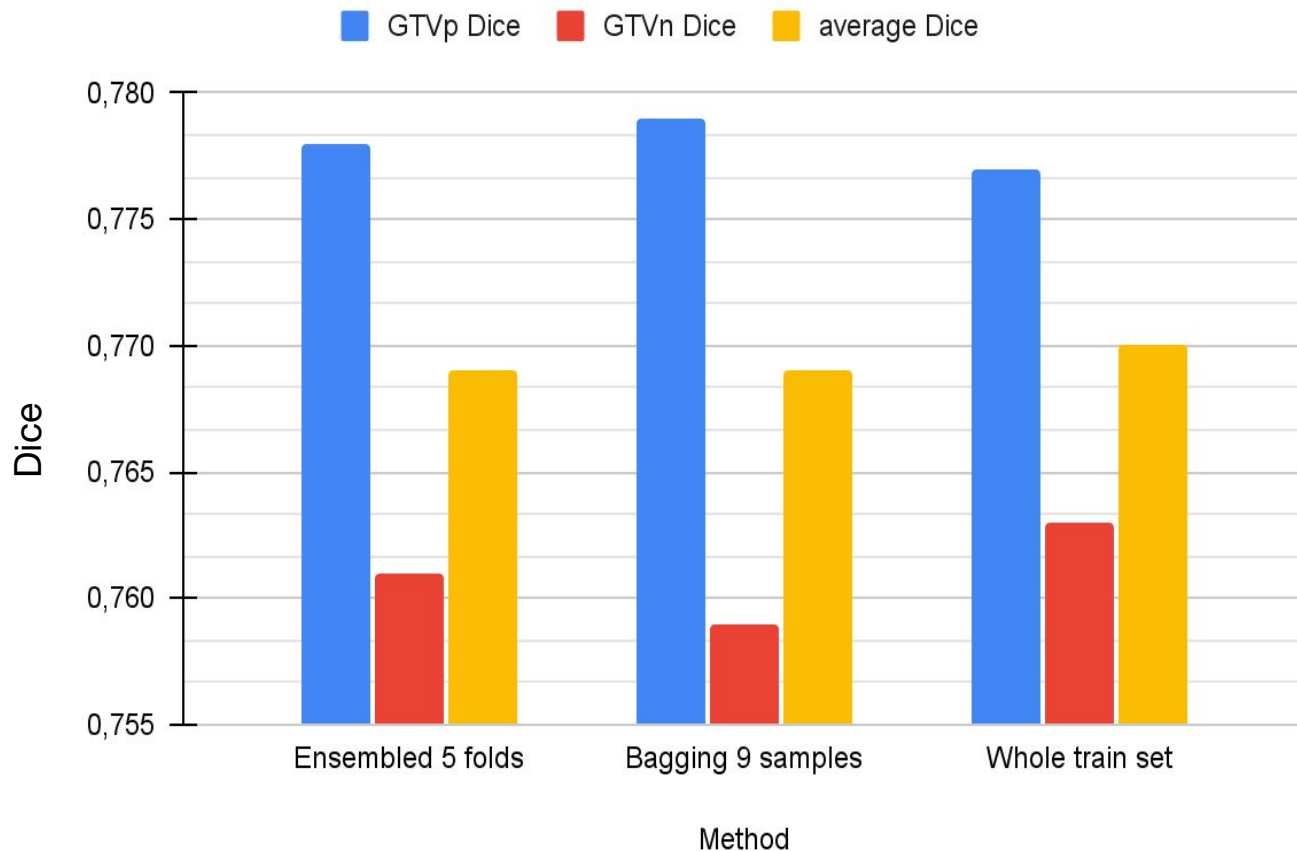
The screenshot shows the GitHub repository for MIC-DKFZ / nnUNet. The repository is public and has 71 watches, 1k forks, and 2.9k stars. The main content area displays a list of files and folders, including .github/ISSUE_TEMP..., documentation, nnunet, tests, .gitignore, HI_Logo.png, LICENSE, readme.md, setup.cfg, and setup.py. The readme.md file is selected, showing a recent update from Fabiansensee text 5 hours ago with 759 reactions. The update text states: "[2020_10_21] Update: We now have documentation for common questions and common issues. We now also provide reference epoch times for several datasets and tips on how to identify bottlenecks. Please read these documents before opening a new issue!". The repository also includes a table of files and folders, a section for releases (9 tags), and a section for packages (No packages published). The repository is used by 52 users and has 30 contributors.

File/Folder	Description	Time Ago
.github/ISSUE_TEMP...	update issue template	2 years ago
documentation	text	5 hours ago
nnunet	add amos2022 code	5 hours ago
tests	typos	5 months ago
.gitignore	initial commit	3 years ago
HI_Logo.png	update logo	6 months ago
LICENSE	setup.py	3 years ago
readme.md	small modifications of documentation	last month
setup.cfg	added more setup stuff	3 years ago
setup.py	resolve unclear pytorch instructions	6 months ago

Segmentation (Task 1) - Results: Test set

Results of the three methods used on the test:

- 5-fold CV averaged
- 9-sample bagging averaged
- 1 model trained on the whole dataset

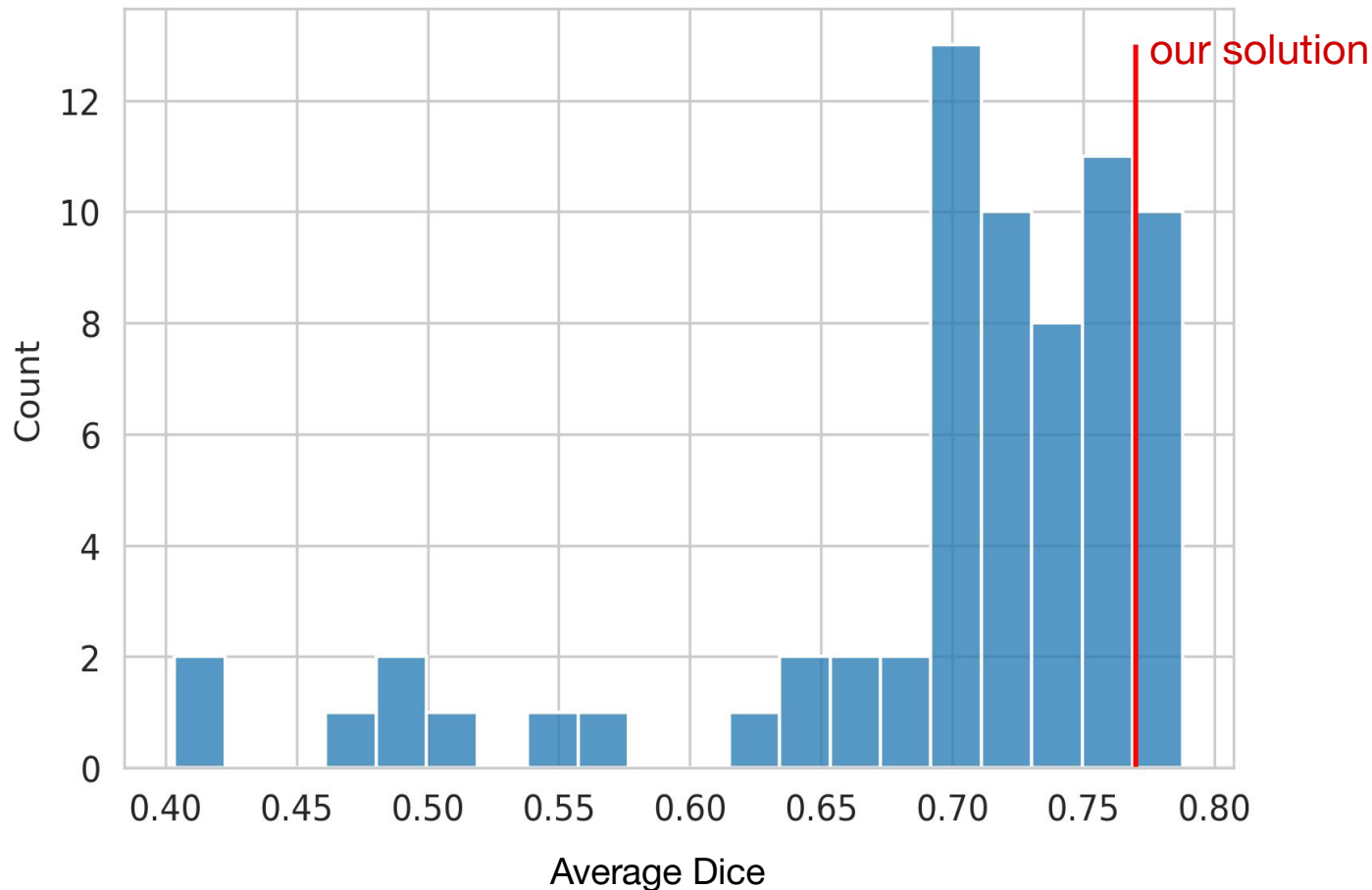


Segmentation (Task 1) - Results: other submissions

Our solution:

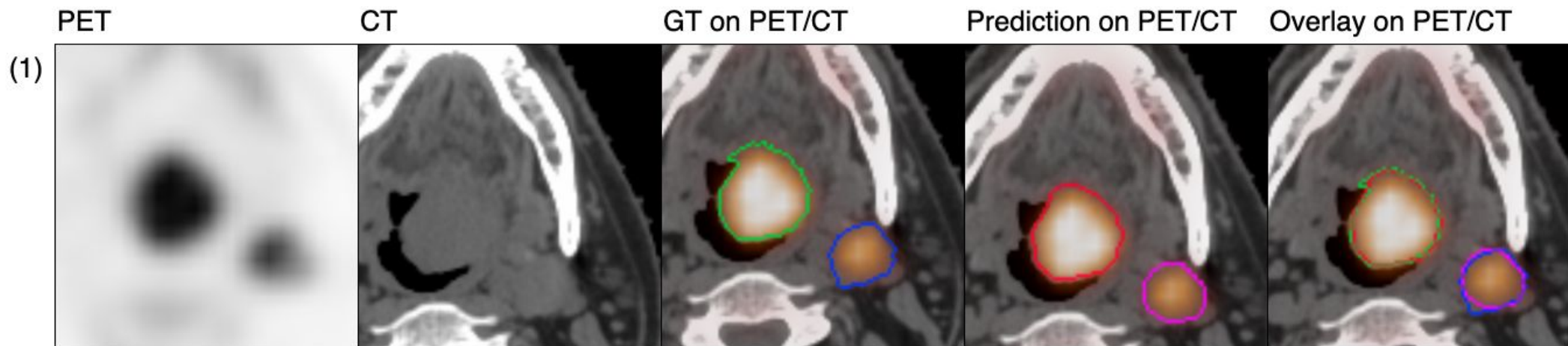
- average Dice: 0.770
- GTVp Dice: 0.777
- GTVn Dice: 0.763

Ranked **4th** at the
HECKTOR challenge



Segmentation (Task 1) - Results

- High Dice scoring patients (average Dice 0.922)



Tumor ground truth



Tumor predicted



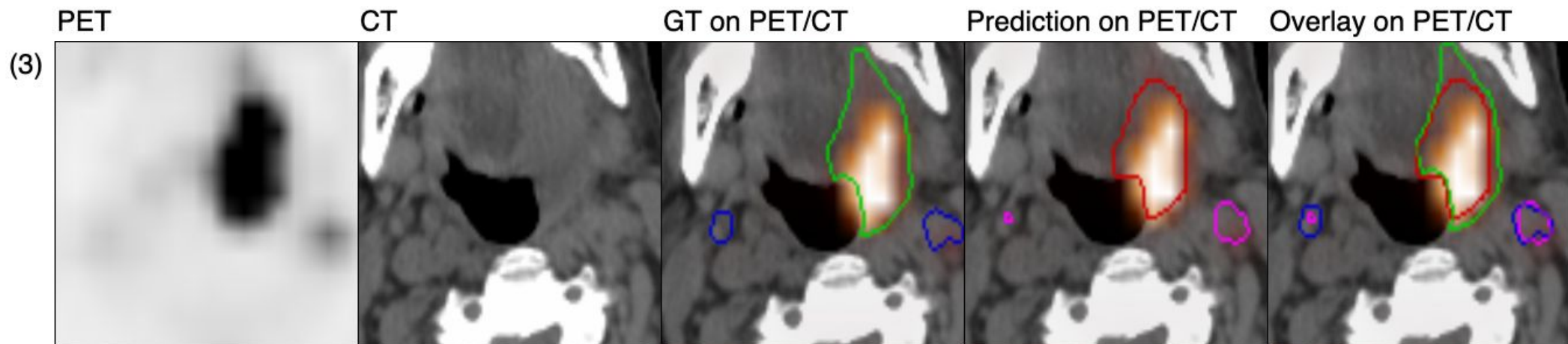
Lymph node ground truth



Lymph node predicted

Segmentation (Task 1) - Results

- Average Dice scoring patients (average Dice 0.761)



Tumor ground truth



Tumor predicted



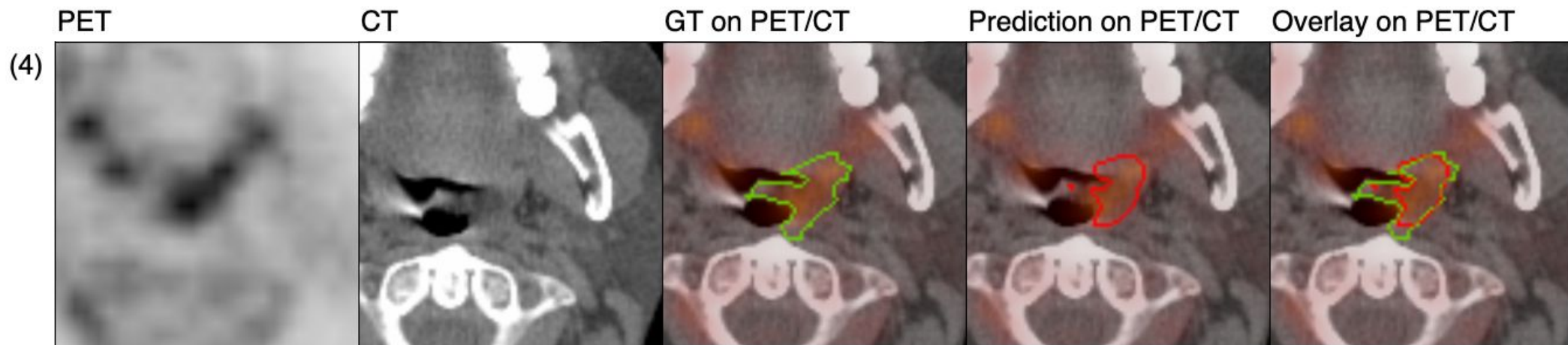
Lymph node ground truth



Lymph node predicted

Segmentation (Task 1) - Results

- Low Dice scoring patients (average Dice 0.303) **due to segmentation errors**



Tumor ground truth



Tumor predicted



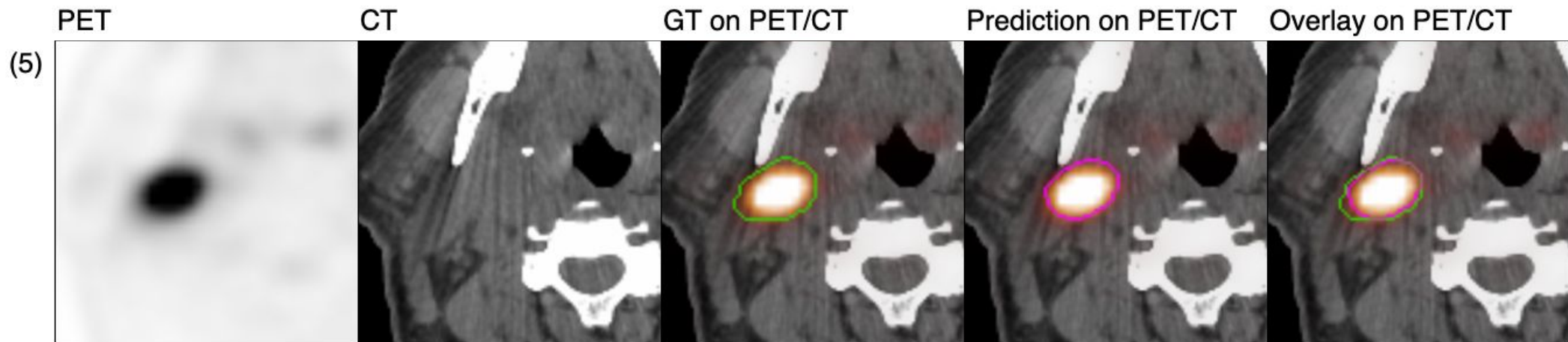
Lymph node ground truth



Lymph node predicted

Segmentation (Task 1) - Results

- Low Dice scoring patients (average Dice 0.000) **due to wrong classification**



Tumor ground truth



Tumor predicted



Lymph node ground truth



Lymph node predicted

Outcome prediction (Task 2) - Binary-weighted model

- It is **hard** to tell if a biomarker is **more important than another** based on limited data
- In survival prediction tasks, we **might not have enough data** to determine **accurate feature importance**
- How does a **model using binary weights only** perform compared to one that learns feature importance?
- To reduce even more the risk of overfitting, **no feature interaction** is learned
- Only the **bare minimum is learned** (minimal learning)

Outcome prediction (Task 2) - Binary-weighted model

- To do so, we **only assign a sign** to each feature, based on their **C-index**:

$$c_i = C_{index}(x_i, y)$$

$$s_i = \begin{cases} +1, & \text{if } c_i \geq 0.5 \\ -1, & \text{otherwise} \end{cases}$$

- The prediction is made by taking the average of the signed normalized features

$$z_i = \frac{x_i - \mu_i}{\sigma_i}$$

$$\hat{y} = \frac{1}{M} \sum_i^M s_i \times z_i$$



Outcome prediction (Task 2) - Binary-weighted model

- Before:
 - we remove features highly correlated to other features (threshold ρ)
- Then, we **only assign a sign** to each feature, based on their **C-index**:

$$c_i = C_{index}(x_i, y) \qquad s_i = \begin{cases} +1, & \text{if } c_i \geq 0.5 \\ -1, & \text{otherwise} \end{cases}$$

- The prediction is made by taking the average of the signed normalized features

$$z_i = \frac{x_i - \mu_i}{\sigma_i} \qquad \hat{y} = \frac{1}{M} \sum_i^M s_i \times z_i$$



Outcome prediction (Task 2) - Binary-weighted model

- Before:
 - we remove features highly correlated to other features (threshold ρ)
 - we drop features with:

$$|c_i| < C_{min} \quad \text{where} \quad |c_i| = \max\{1 - c_i, c_i\}$$

- Then, we **only assign a sign** to each feature, based on their **C-index**:

$$c_i = C_{index}(x_i, y) \qquad s_i = \begin{cases} +1, & \text{if } c_i \geq 0.5 \\ -1, & \text{otherwise} \end{cases}$$

- The prediction is made by taking the average of the signed normalized features

$$z_i = \frac{x_i - \mu_i}{\sigma_i} \qquad \hat{y} = \frac{1}{M} \sum_i s_i \times z_i$$



Outcome prediction (Task 2) - Binary-weighted model

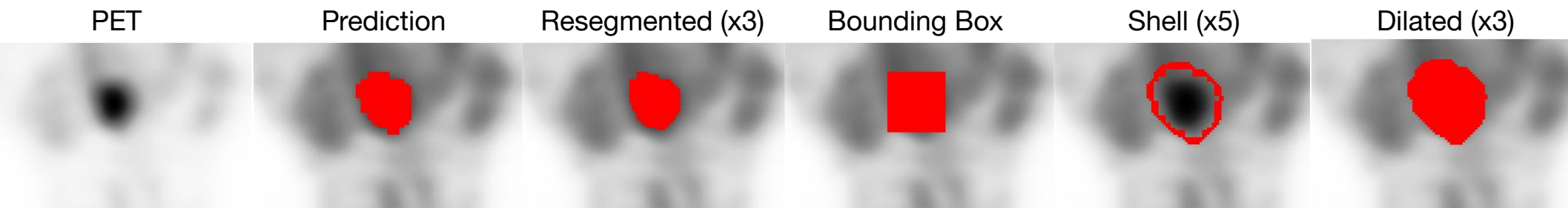
- Hypothesis: such model is **resilient to the curse of dimensionality** and can only benefit from additional features
→ a lot of features are needed
- To reduce even more the overfitting and produce better predictions, a **bagging** strategy was adopted for the challenge (with a median aggregator)

Outcome prediction (Task 2) - Binary-weighted model

- This algorithm has some hyperparameters:
 - collinearity removal: Pearson correlation threshold
 - minimum absolute distance from 0.5 C-index
 - number of randomly selected features
- Random search optimized



Outcome prediction (Task 2) - Radiomic features

- To make the model resilient to **segmentation variations** and to produce more features, we built 13 masks the following types:



- For each mask, we computed features with **pyradiomics**:
 - 93 on the PET images
 - 93 on the CT images
- 3 additional handcrafted features (tumors and lymph nodes number, whole-body or H&N scan/assessment)
- Produced **2421 features, 2430** when including the **clinical** ones

Outcome prediction (Task 2) - Results

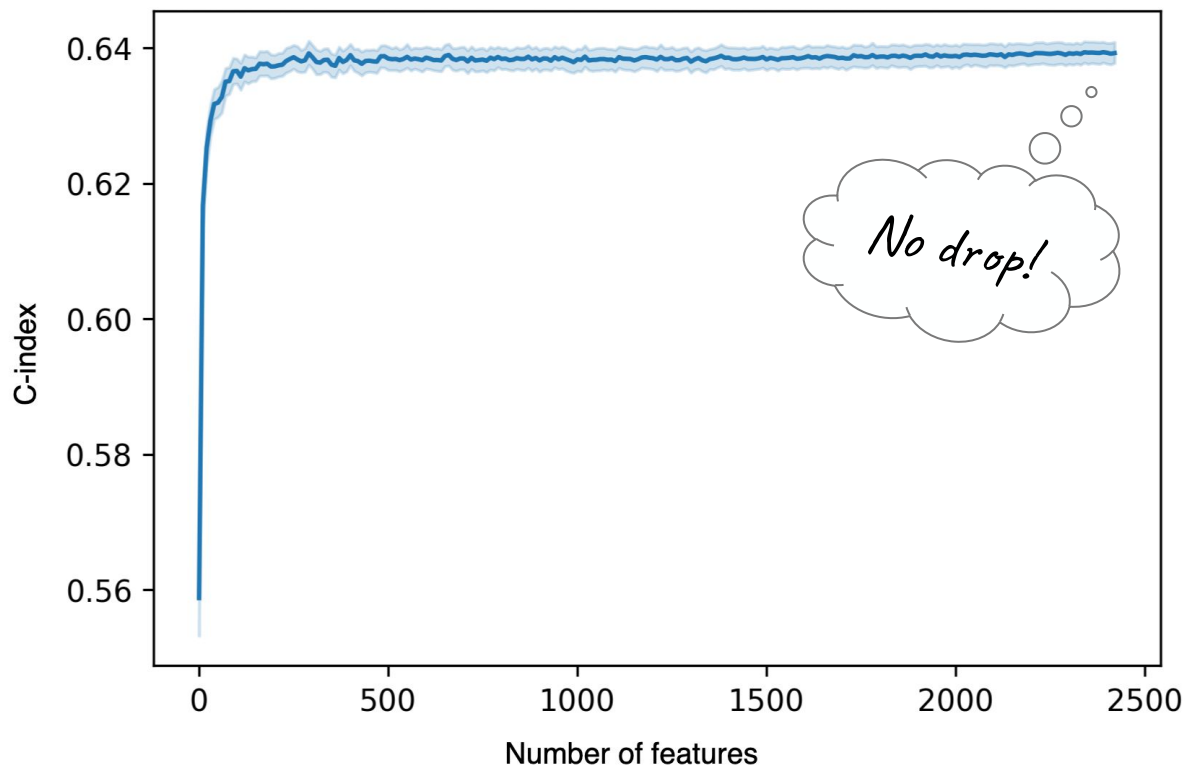
Model	CV C-index train set (CI)	C-index test set	hyperparameters sets tested (#)
Binary-weighted	0.645 (0.585 - 0.707)		10
Binary-weighted bagged	0.668 (0.605 - 0.730)	0.670	10
Binary-weighted bagged	0.675 (0.613 - 0.731)	0.673	100
Binary-weighted bagged	0.688 (0.642 - 0.732)	 0.682 	1000

- Remark: for equal hyperparametrization strength (10 here), **bagging improved results**



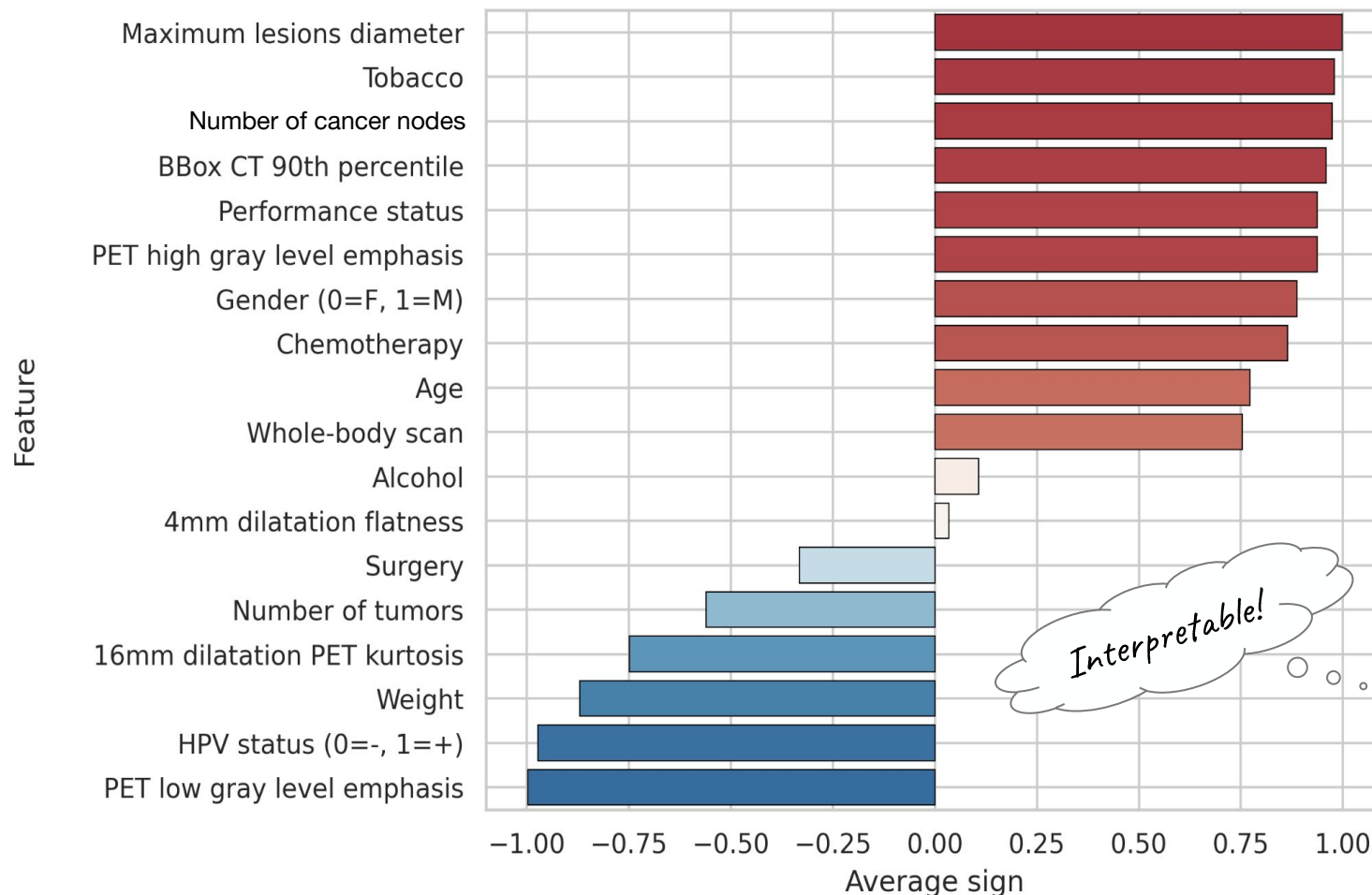
Outcome prediction (Task 2) - Results: curse of dimensionality

- Evaluation of a single model on the train set with a Monte Carlo CV



Outcome prediction (Task 2) - Results

Best model submitted
average features signs:



Outcome prediction (Task 2) - Results

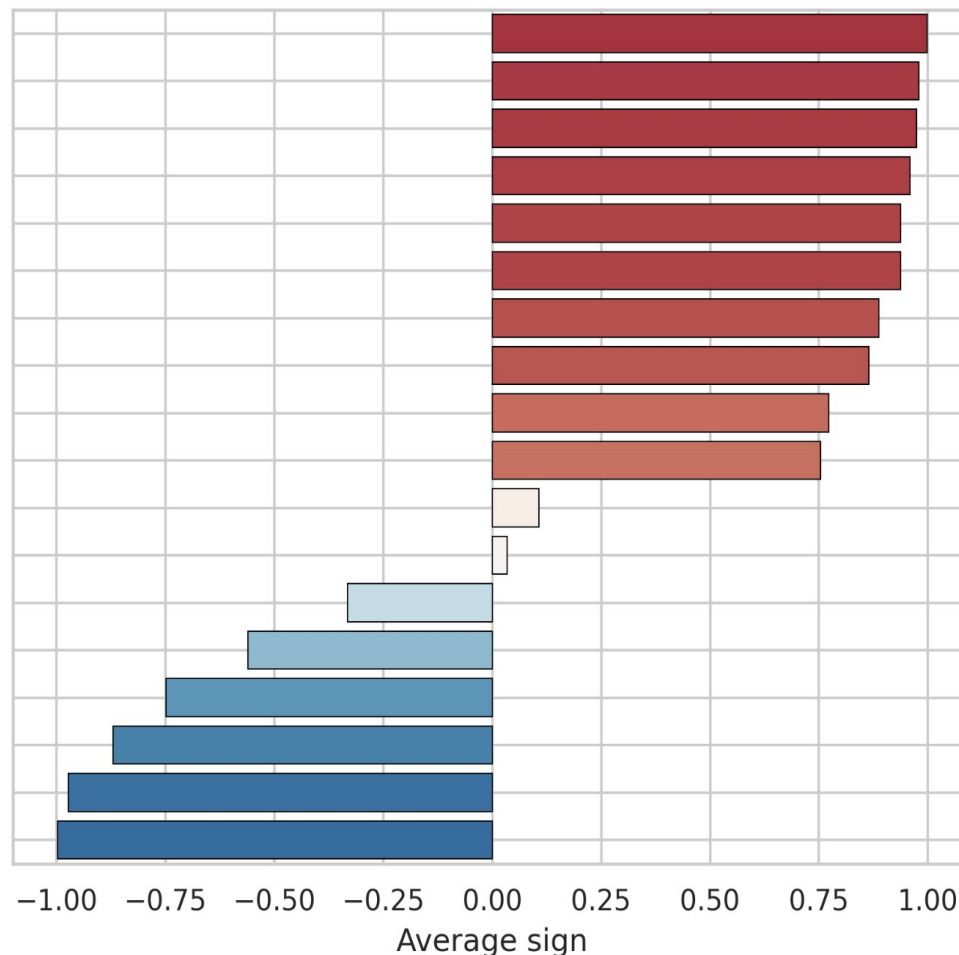
Best model submitted
average features signs:

High number of
missing values



Feature

Maximum lesions diameter
Tobacco
Number of cancer nodes
BBox CT 90th percentile
Performance status
PET high gray level emphasis
Gender (0=F, 1=M)
Chemotherapy
Age
Whole-body scan
Alcohol
4mm dilatation flatness
Surgery
Number of tumors
16mm dilatation PET kurtosis
Weight
HPV status (0=-, 1=+)
PET low gray level emphasis



Outcome prediction (Task 2) - GitHub repository

- The model is publicly available on GitHub and pip as ICARE (Individual Coefficient Approximation for Risk Estimation)

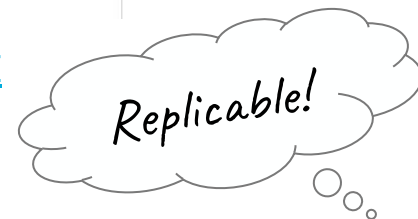
```
$ pip install icare
```

- It is fully scikit-compatible
- The code and the trained model full pipeline will be published soon.



A screenshot of the GitHub repository page for "Lrebaud/ICARE". The page shows the repository name, a table of files, and the README content. The file table lists files like "icare", "img", "notebook", "test", ".gitignore", "LICENSE", "README.md", "setup.py", and "tox.ini" with their commit times. The README section includes a "pypi package" badge for version 0.0.3, the repository name "ICARE", a description of the model, and a list of contributors. The right sidebar shows repository statistics like stars, forks, and languages.

<https://github.com/Lrebaud/ICARE>



Conclusion

- **Fully automatic pipeline** producing **accurate tumor and lymph nodes segmentation** and state-of-the-art **prediction of the recurrence risk**
- A **straightforward** approach for segmentation was **competitive** with task-specialized ones
- **Reducing learning** to its bare minimum produced the **best performance** for outcome prediction
- We propose a **novel, simple and robust** model delivering accurate risk prediction
- The number of cancer nodes was of prognostic value, showing the importance of **GTVn segmentation and lymph nodes assessment**



Acknowledgements



Thank you for your attention!

We are hiring!

PhD students, post-doc, engineers and researchers,
medical physicists, radiobiologists, radiopharmacists,
and data scientists

Ir ne Buvat
U1288 LITO
Institut Curie, Inserm
Orsay, France

irene.buvat@curie.fr
 [@irene.buvat](https://twitter.com/irene.buvat)

www.lito-web.fr
www.irene.guillemet.org